Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A digital processing system comprising a microprocessor, wherein the microprocessor is operable to perform a method for calling a subroutine, the method comprising the steps of:

branching to the subroutine by executing a first instruction to provide an address of the subroutine; and

calculating a complete return address by executing a second instruction that is ~~sequentially adjacent~~ next to the first instruction to determine a relative return address, wherein no instructions separate the first and second instructions.

2. (Currently Amended) The digital processing system of Claim 1, wherein the step of calculating a complete return address comprises the step of adding a relative displacement value provided by the second instruction to a program address value associated with the second instruction.

3. (Currently Amended) The digital processing system of Claim 2, wherein the step of calculating a complete return address further comprises the step of storing the return address in a general purpose register of the microprocessor.

4. (Original) The digital processing system of Claim 1, wherein the second instruction is executed during a delay slot associated with the first instruction.

5. (Original) The digital processing system of Claim 1, wherein the second instruction is executed before executing the first instruction.

6. (Currently Amended) The digital processing system of Claim 1, wherein during the step of calculating a complete return address, a plurality of second instructions are

conditionally executed in response to a predicate value such that the return address is responsive to the predicate value.

7. (Currently Amended) The digital processing system of Claim 1, wherein during the step of calculating a complete return address the return address is remotely associated with the second instruction.

8. (Original) The digital processing system of Claim 1, further comprising the step of filling a number of delay slots associated with the first instruction in an instruction execution pipeline of the microprocessor by executing a number of virtual no-operation (NOP) instructions specified by the second instruction.

9. (Currently Amended) A digital processing system comprising a microprocessor, wherein the microprocessor is operable to perform a method for forming a relative return address, the method comprising the steps of:

fetching a sequence of instructions in response to address locations provided by a program counter; and

executing a first instruction immediately after executing a branch-to-subroutine instruction in the sequence of instructions by using a first address value provided by the program counter as a source operand, wherein said first instruction forms the relative return address, said relative return address associated with the branch-to-subroutine instruction.

10. (Currently Amended) The digital system of Claim 9, wherein the step of executing a first instruction comprises the step of combining a displacement value provided by the first instruction with the first address value provided by the program counter to calculate the relative return address.

11. (Original) The digital system of Claim 10, wherein the step of executing the first instruction further comprises the step of providing a number of virtual no-operation (NOP) instructions for execution after the step of executing the first instruction.

12. (Original) The digital system of Claim 10, wherein the microprocessor comprises:

an instruction execution pipeline having a plurality of stages;

the program counter being associated with a fetch stage of the instruction execution pipeline;

FIFO circuitry connected to receive address values from the program counter, the FIFO circuitry operable to delay each address value received from the program counter; and

a first functional circuit associated with an execution phase of the instruction execution pipeline connected to receive a delayed address value from the FIFO circuitry, wherein the first functional circuit is operable to add the displacement value provided by the first instruction to a first delayed address value provided by the program counter.

13. (Previously Presented) The digital system of Claim 12, further comprising:

a plurality of functional units associated with the execution phase of the instruction execution pipeline;

wherein the instruction execution pipeline receives a fetch packet containing a plurality of instructions associated with each address value provided by the program counter;

wherein a dispatch stage of the pipeline is operable to provide an execution packet that spans two or more fetch packets; and

wherein the FIFO circuitry is operable to provide the first delayed address value such that the first delayed address value is associated with one of the two or more fetch packets in which an instruction executed by the first functional unit is located.

14. (Previously Presented) The digital system according to Claim 9 being a cellular telephone, further comprising:

an integrated keyboard connected to the microprocessor via a keyboard adapter;

a display, connected to the microprocessor via a display adapter;

radio frequency (RF) circuitry connected to the microprocessor; and

an aerial connected to the RF circuitry.

15. (Currently Amended) A method for calling a subroutine in a digital processing system comprising a microprocessor, the method comprising the steps of:

branching to the subroutine by executing a first instruction to provide an address of the subroutinc; and

calculating a complete return address by executing a second instruction that is ~~sequentially adjacent to~~ next to the first instruction to determine a relative return address, wherein no instructions separate the first and second instructions.

16. (Original) The method of Claim 15, wherein the step of calculating a return address comprises the step of adding a relative displacement value provided by the second instruction to a program address value associated with the second instruction.

17. (Previously Presented) The method of Claim 15, wherein the step of calculating a return address further comprises the step of storing the return address in a general purpose register of the microprocessor.

18. (Currently Amended) A method for forming a relative return address in a digital processing system comprising a microprocessor, the method comprising the steps of:

fetching a sequence of instructions in response to address locatrons provided by a program counter; and

executing a first instruction immediately after executing a branch-to-subroutine instruction in the sequence of instructions by using a first address value provided by the program counter as a source operand, wherein said first instruction forms the relative return address, said relative return address associated with the branch-to-subroutine instruction.

19. (Original) The method of Claim 18, wherein the step of executing a first instruction comprises the step of adding a displacement value provided by the first instruction to the first address value provided by the program counter.

20. (Previously Presented) The method of Claim 19, wherein the step of executing the first instruction further comprises the step of providing a number of virtual no-operation (NOP) instructions for execution after the step of executing the first instruction.

21. (Currently Amended) A method for operating a compiler for a microprocessor, wherein the microprocessor is operable to execute a first type of instruction that performs a specified operation and also directs that a selectable number of virtual no-operation (NOP) instructions be executed after executing the first type instruction, the method comprising the steps of:

determining that a first instruction of the first type of instruction is to be executed in a delay slot of a first branch type instruction, wherein said first instruction calculates a ~~first~~ complete return address;

determining that a second instruction of a second type of instruction is to be executed in a delay slot of the first branch type instruction~~, wherein said second instruction calculates a second return address~~;

arranging an instruction sequence such that the first instruction is executed after the second instruction;

determining a remaining number of unused delay slots of the first branch type instruction; and

inserting a value in the first instruction to direct that a selectable number of virtual NOP instructions be executed after executing the first type instruction, wherein the selectable number of NOPs is the remaining number of unused delay slots of the first branch instruction.